



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/934,407	08/21/2001	Cary Lee Bates	ROC9200110085US1	8109

7590 02/22/2005

Gero G. McClellan
Thomason, Moser & Patterson, L.L.P.
Suite 1500
3040 Post Oak Boulevard
Houston, TX 77056-6582

EXAMINER

YIGDALL, MICHAEL J

ART UNIT	PAPER NUMBER
----------	--------------

2122

DATE MAILED: 02/22/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary

Application No.

09/934,407

Applicant(s)

BATES ET AL.

Examiner

Michael J. Yigdall

Art Unit

2122

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 01 November 2004.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-26 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-26 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
- ☐ Certified copies of the priority documents have been received.
 - ☐ Certified copies of the priority documents have been received in Application No. _____.
 - ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|--|---|
| 1) <input type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152) |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____ |

DETAILED ACTION

1. Applicant's response and amendment filed on November 1, 2004 has been fully considered. Claims 1-26 are pending.
2. Applicant notes that "claims 26 remain pending" and submits that "new claims do not introduce new matter" (Applicant's remarks, page 7, paragraph 3). However, no new claims have been added by the response, and all claims 1-26 are presented as originally filed. Clarification of Applicant's statement is requested.

Response to Arguments

3. Applicant's arguments have been fully considered but they are not persuasive.
4. Applicant contends that Olsen does not disclose, "a method for inserting breakpoints in a program being debugged" (Applicant's remarks, page 7), as recited in the preamble of claim 1.

However, Olsen expressly discloses a method for debugging a program (see, for example, the abstract) that includes the step of setting or inserting breakpoints in the program (see, for example, step 50 in FIG. 4a). When the user sets or inserts a breakpoint, the debugger likewise sets or inserts a corresponding breakpoint in the machine code (see, for example, column 5, lines 29-31), as Applicant acknowledges (Applicant's remarks, page 8). Olsen, therefore, discloses a method for inserting breakpoints in a program being debugged.
5. Applicant contends that Olsen does not disclose, "determining which blocks control execution of the basic block" (Applicant's remarks, page 8), as recited in claims 1, 12 and 16.

However, Olsen discloses analyzing the paths of execution leading to the basic block (see, for example, column 12, lines 40-45). A path of execution is comprised of other basic blocks (see, for example, column 12, lines 28-32). The blocks that form the path must be determined in order to analyze the path. Because the analysis is based on control flow (see, for example, column 12, lines 20-24), Olsen discloses determining which blocks control execution of the basic block.

Moreover, Olsen expressly discloses determining which instructions affect or control the execution of a branch (see, for example, column 7, line 65 to column 8, line 27), which is to say determining which basic blocks affect or control the execution of another basic block (see, for example, column 5, lines 60-62).

6. Applicant contends that because Olsen does not disclose determining which blocks control execution of the basic block, the techniques disclosed in Shagam lack the information necessary to determine where to insert any breakpoints into the program being debugged, and the combination is thus inoperable.

However, Olsen does disclose determining which blocks control execution of the basic block, as discussed above. Shagam then discloses inserting a trace point at each block of execution (see, for example, column 6, lines 23-28). In other words, Shagam discloses inserting a breakpoint (see, for example, column 1, lines 20-25) at each control flow boundary in the blocks of execution (see, for example, column 2, lines 9-31). A branch is an example of a control flow boundary. Therefore, Olsen in view of Shagam discloses inserting a breakpoint at each branch contained in the blocks controlling execution of the basic block.

Claim Rejections - 35 USC § 103

7. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

8. Claims 1-26 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Pat. No. 6,263,489 to Olsen et al. (art of record, "Olsen") in view of U.S. Pat. No. 6,161,216 to Shagam (art of record, "Shagam").

With respect to claim 1 (original), Olsen discloses a method for inserting breakpoints in a program being debugged (see, for example, the abstract, and step 50 in FIG. 4a), comprising:

(a) identifying a statement for the program being debugged (see, for example, column 12, lines 33-39, which shows identifying a statement);

(b) determining which basic block contains the statement (see, for example, column 12, lines 28-32, which shows determining the basic block from which the statement is identified);

(c) determining which blocks control execution of the basic block (see, for example, column 12, lines 20-24 and 40-45, which shows analyzing the paths of execution leading to the basic block based on control flow, i.e. determining which blocks control execution of the basic block; also see, for example, column 7, line 65 to column 8, line 27, which shows determining which instructions affect or control execution of a branch, i.e. determining which blocks control execution of another basic block).

Art Unit: 2122

Although Olsen discloses inserting commit points for all possible breakpoint locations (see, for example, column 8, lines 37-42) and determining exit points for the basic blocks (see, for example, column 10, lines 17-25), Olsen does not expressly disclose:

(d) inserting a breakpoint at each branch contained in the blocks controlling execution of the basic block.

However, Shagam discloses inserting trace points at each block of execution (see, for example, column 6, lines 23-28), or in other words, at each control flow boundary in the blocks of execution (see, for example, column 2, lines 9-31). A branch is an example of a control flow boundary. The trace points are analogous to breakpoints (see, for example, column 1, lines 9-13). Shagam discloses that the method for inserting the trace points or breakpoints improves the performance and accuracy of debugging (see, for example, column 1, lines 29-36 and 57-62).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to supplement the method of Olsen with the step of inserting a breakpoint at each branch contained in the blocks controlling execution of the basic block, so as improve the performance and accuracy of the debugging, such as taught by Shagam.

With respect to claim 2 (original), Olsen also discloses the limitation wherein the statement is the location where the program being debugged halted execution (see, for example, column 13, lines 13-20, which shows identifying the instruction or statement at which the program suspended or halted execution).

With respect to claim 3 (original), Olsen also discloses the limitation wherein the blocks controlling execution of the basic block are blocks on which the basic block is control dependent

Art Unit: 2122

(see, for example, column 5, line 46 to column 6, line 12, which shows that blocks in the path of execution of the basic block are dependent upon control flow).

With respect to claim 4 (original), Olsen also discloses the limitation wherein identifying the statement comprises identifying a statement that may modify a program variable (see, for example, column 7, lines 1-11, which shows identifying statements that may modify a variable).

With respect to claim 5 (original), Olsen also discloses the limitation wherein identifying the statement that may modify a program variable comprises accessing a table comprising the variable mapped to statements that may modify the variable (see, for example, column 6, lines 55-67, which shows a range table for mapping the variable to statements that may modify the variable).

With respect to claim 6 (original), Olsen also discloses the limitation wherein identifying the statement comprises identifying statements associated with loop latches (see, for example, column 13, lines 48-60, which shows identifying statements associated with loops).

With respect to claim 7 (original), Olsen also discloses the limitation wherein identifying statements associated with loop latches comprises accessing tables comprising the basic block mapped to the loop latches (see, for example, column 8, lines 51-61, which shows a table for mapping source loops to basic blocks in the machine code).

With respect to claim 8 (original), Olsen also discloses the limitation wherein identifying the statement comprises identifying a currently executing statement of each of a plurality of

Art Unit: 2122

subprograms (see, for example, column 7, lines 15-47, which shows identifying and executing statements of a plurality of functions, i.e. subprograms).

With respect to claim 9 (original), Olsen also discloses the limitation wherein each of the plurality of subprograms is a portion of the program being debugged and performs a specific task (see, for example, column 7, lines 15-47, which shows that the functions or subprograms in the program are debugged, and that the functions change the state of the program, i.e. perform a task).

With respect to claim 10 (original), Olsen also discloses the limitation wherein identifying the currently executing statement comprises accessing a table comprising the plurality of subprograms mapped to its respective currently executing statement (see, for example, column 8, lines 43-50, which shows a table for mapping source constructs, e.g. subprograms, to machine code instructions that are executed).

With respect to claim 11 (original), Olsen also discloses the limitation wherein the blocks controlling execution of the basic block are blocks on which the basic block is control dependent (see, for example, column 5, line 46 to column 6, line 12, which shows that blocks in the path of execution of the basic block are dependent upon control flow).

With respect to claim 12, Olsen discloses a computer system comprising at least one processor configured to execute a debugging program (see, for example, CPU 12 and debugger 30 of computer system 10 in FIG. 1), wherein the processor, when executing the debugging program, is configured to perform an operation (see, for example, the abstract) comprising:

(a) identifying a statement for the program being debugged (see, for example, column 12, lines 33-39, which shows identifying a statement);

(b) determining which basic block contains the statement (see, for example, column 12, lines 28-32, which shows determining the basic block from which the statement is identified);

(c) determining which blocks control execution of the basic block (see, for example, column 12, lines 20-24 and 40-45, which shows analyzing the paths of execution leading to the basic block based on control flow, i.e. determining which blocks control execution of the basic block; also see, for example, column 7, line 65 to column 8, line 27, which shows determining which instructions affect or control execution of a branch, i.e. determining which blocks control execution of another basic block).

Although Olsen discloses inserting commit points for all possible breakpoint locations (see, for example, column 8, lines 37-42) and determining exit points for the basic blocks (see, for example, column 10, lines 17-25), Olsen does not expressly disclose:

(d) inserting a breakpoint at each branch contained in the blocks controlling execution of the basic block.

However, Shagam discloses inserting trace points at each block of execution (see, for example, column 6, lines 23-28), or in other words, at each control flow boundary in the blocks of execution (see, for example, column 2, lines 9-31). A branch is an example of a control flow boundary. The trace points are analogous to breakpoints (see, for example, column 1, lines 9-13). Shagam discloses that the method for inserting the trace points or breakpoints improves the performance and accuracy of debugging (see, for example, column 1, lines 29-36 and 57-62).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to supplement the method of Olsen with the step of inserting a breakpoint at each branch contained in the blocks controlling execution of the basic block, so as improve the performance and accuracy of the debugging, such as taught by Shagam.

With respect to claim 13 (original), Olsen also discloses the limitation wherein identifying the statement comprises:

(a) identifying a program variable (see, for example, column 6, lines 55-67, which shows identifying a variable); and

(b) determining which statements may modify the variable (see, for example, column 7, lines 1-11, which shows determining statements that may modify the variable).

With respect to claim 14 (original), Olsen also discloses the limitation wherein identifying the statement comprises:

(a) determining a plurality of sets of loop latches for the basic block (see, for example, column 8, lines 51-61, which shows determining a plurality of loops associated with basic blocks in the machine code); and

(b) identifying the statements associated with loop latches (see, for example, column 13, lines 48-60, which shows identifying statements associated with loops).

With respect to claim 15 (original), Olsen further discloses the limitation wherein identifying the statement comprises identifying a currently executing statement of a plurality of subprograms (see, for example, column 7, lines 15-47, which shows identifying and executing statements of a plurality of functions, i.e. subprograms).

With respect to claim 16 (original), Olsen discloses a signal-bearing medium (see, for example, medium 36 in FIG. 1), comprising a program (see, for example, debugger 30 in FIG. 1) which, when executed by a processor (see, for example, CPU 12 in FIG. 1), performs an operation (see, for example, the abstract), comprising:

(a) identifying a statement for the program being debugged (see, for example, column 12, lines 33-39, which shows identifying a statement);

(b) determining which basic block contains the statement (see, for example, column 12, lines 28-32, which shows determining the basic block from which the statement is identified);

(c) determining which blocks control execution of the basic block (see, for example, column 12, lines 20-24 and 40-45, which shows analyzing the paths of execution leading to the basic block based on control flow, i.e. determining which blocks control execution of the basic block; also see, for example, column 7, line 65 to column 8, line 27, which shows determining which instructions affect or control execution of a branch, i.e. determining which blocks control execution of another basic block).

Although Olsen discloses inserting commit points for all possible breakpoint locations (see, for example, column 8, lines 37-42) and determining exit points for the basic blocks (see, for example, column 10, lines 17-25), Olsen does not expressly disclose:

(d) inserting a breakpoint at each branch contained in the blocks controlling execution of the basic block.

However, Shagam discloses inserting trace points at each block of execution (see, for example, column 6, lines 23-28), or in other words, at each control flow boundary in the blocks of execution (see, for example, column 2, lines 9-31). A branch is an example of a control flow

Art Unit: 2122

boundary. The trace points are analogous to breakpoints (see, for example, column 1, lines 9-13). Shagam discloses that the method for inserting the trace points or breakpoints improves the performance and accuracy of debugging (see, for example, column 1, lines 29-36 and 57-62).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to supplement the method of Olsen with the step of inserting a breakpoint at each branch contained in the blocks controlling execution of the basic block, so as improve the performance and accuracy of the debugging, such as taught by Shagam.

With respect to claim 17 (original), the limitations recited in the claim correspond to the limitations recited in claim 2 (see the explanation for claim 2 set forth above).

With respect to claim 18 (original), the limitations recited in the claim correspond to the limitations recited in claim 3 (see the explanation for claim 3 set forth above).

With respect to claim 19 (original), the limitations recited in the claim correspond to the limitations recited in claim 4 (see the explanation for claim 4 set forth above).

With respect to claim 20 (original), the limitations recited in the claim correspond to the limitations recited in claim 5 (see the explanation for claim 5 set forth above).

With respect to claim 21 (original), the limitations recited in the claim correspond to the limitations recited in claim 6 (see the explanation for claim 6 set forth above).

With respect to claim 22 (original), the limitations recited in the claim correspond to the limitations recited in claim 7 (see the explanation for claim 7 set forth above).

With respect to claim 23 (original), the limitations recited in the claim correspond to the limitations recited in claim 8 (see the explanation for claim 8 set forth above).

With respect to claim 24 (original), the limitations recited in the claim correspond to the limitations recited in claim 9 (see the explanation for claim 9 set forth above).

With respect to claim 25 (original), the limitations recited in the claim correspond to the limitations recited in claim 10 (see the explanation for claim 10 set forth above).

With respect to claim 26 (original), the limitations recited in the claim correspond to the limitations recited in claim 11 (see the explanation for claim 11 set forth above).

Conclusion

9. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Art Unit: 2122

10. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Michael J. Yigdall whose telephone number is (571) 272-3707.

The examiner can normally be reached on Monday through Friday from 7:30am to 4:00pm.

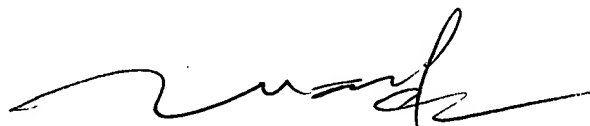
If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Q. Dam can be reached on (571) 272-3695. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

MY

Michael J. Yigdall
Examiner
Art Unit 2122

mjy



TUAN DAM
SUPERVISORY PATENT EXAMINER